Workshop on Essential Abstractions in GCC

A Summary of Essential Abstrations

GCC Resource Center

(www.cse.iitb.ac.in/grc)

Department of Computer Science and Engineering, Indian Institute of Technology, Bombay



3 July 2011

3 July 2011 Essential Abstrations: Summary

3 July 2011

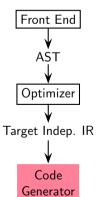
Essential Abstrations: Summary

1/15

Compilation Models

Compilation Models

Aho Ullman Model



Aho Ullman: Instruction selection

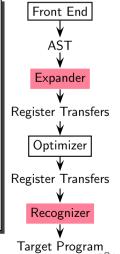
- over optimized IR using
- cost based tree pattern matching

Davidson Fraser: Instruction selection

- over AST using
- structural tree pattern matching
- naive code which is
 - ▶ target dependent, and is
 - optimized subsequently

Davidson Fraser Model

1/15

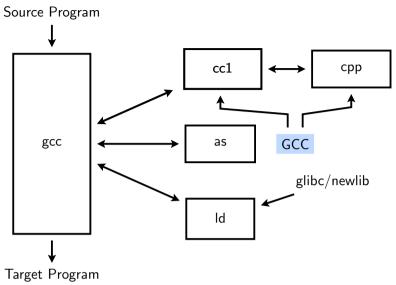


Notes

GCC Resource Center, IIT Bom

Target Program

The GNU Tool Chain



Essential Abstractions in GCC

GCC Resource Center, IIT Bombay

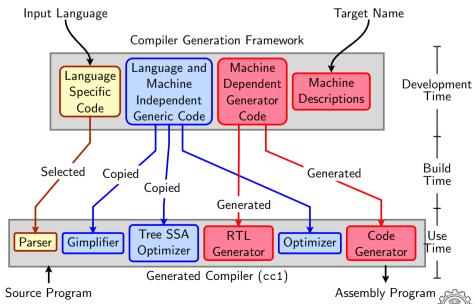


3/15

2/15

3 July 2011 Essential Abstrations: Summary

The Architecture of GCC



The GNU Tool Chain

Notes

Essential Abstractions in GCC

GCC Resource Center, IIT Bombay

3/15

Essential Abstrations: Summary

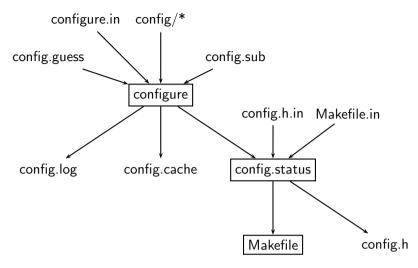
The Architecture of GCC

Notes

3 July 2011



Configuring GCC

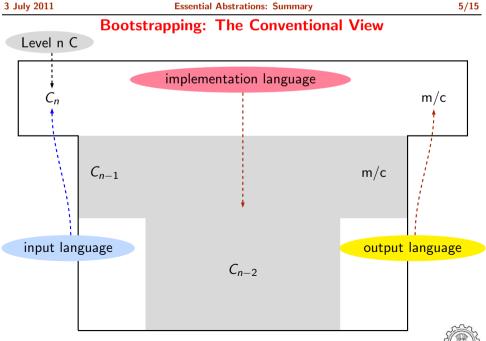


Essential Abstractions in GCC

GCC Resource Center, IIT Bomba



Essential Abstrations: Summary 5/15



Configuring GCC

Essential Abstractions in GCC

GCC Resource Center, IIT Bomba

5/15

3 July 2011

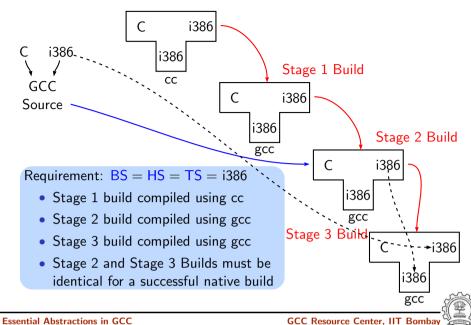
Essential Abstrations: Summary Bootstrapping: The Conventional View



6/15

7/15

A Native Build on i386



3 July 2011

Build for a Given Machine

This is what actually happens!

- Generation
 - Generator sources
 (\$(SOURCE_D)/gcc/gen*.c) are read and
 generator executables are created in
 \$(BUILD)/gcc/build
 - ► MD files are read by the generator executables and back end source code is generated in \$(BUILD)/gcc
- Compilation
 Other source files are read from
 \$(SOURCE_D) and executables created in
 corresponding subdirectories of \$(BUILD)
- Installation
 Created executables and libraries are copied
 in \$(INSTALL)

genattr gencheck genconditions genconstants genflags genopinit genpreds genattrtab genchecksum gencondmd genemit gengenrtl genmddeps genoutput genrecog genautomata gencodes genconfig genextract gengtype genmodes genpeep

A Native Build on i386

Notes

Essential Abstractions in GCC

GCC Resource Center, IIT Bombay

ay Free free

3 July 2011

Essential Abstrations: Summary

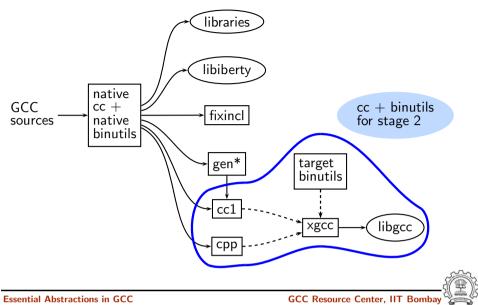
7/15

Build for a Given Machine

Notes



More Details of an Actual Stage 1 Build for C



3 July 2011

Essential Abstrations: Summary

3 July 2011

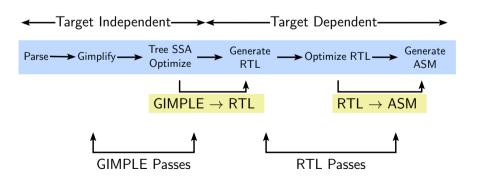
Essential Abstrations: Summary

9/15

Basic Transformations in GCC

Basic Transformations in GCC

Tranformation from a language to a different language





9/15

More Details of an Actual Stage 1 Build for C

Notes

Essential Abstractions in GCC

GCC Resource Center, III Bonibay

GCC GCC Resource Center, IIT Bombay

lote

Essential Abstrations: Summary Instruction Specification and Translation: A Recap

```
Target Independent

Parse Gimplify Tree SSA Optimize RTL

Generate RTL

Generate RTL

GIMPLE: target independent

RTL: target dependent

Need: associate the semantics

GCC Solution: Standard Pattern Names

GIMPLE_ASSIGN

RTL Template

ASM

GIMPLE_ASSIGN

(define_insn "movsi"

(match_operand 0 "register_operand" "r")

(match_operand 1 "const_int_operand" "k"))]

"" /* C boolean expression, if required */

"li %0, %1"

()
```

3 July 2011 Essential Abstrations: Summary

Essential Abstractions in GCC

11/15

Translation Sequence in GCC

```
(define_insn
   "movsi"
   [(set
        (match_operand 0 "register_operand" "r")
        (match_operand 1 "const_int_operand" "k")
   )]
   "" /* C boolean expression, if required */
   "li %0, %1"
)
```

```
D.1283 = 10;  

(set (reg:SI 58 [D.1283]) (const_int 10: [0xa]) 
)  

1i $t0, 10
```

Instruction Specification and Translation: A Recap

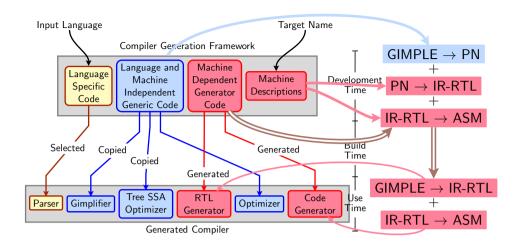
Notes

3 July 2011 Essential Abstrations: Summary 11/15

Translation Sequence in GCC

Notes

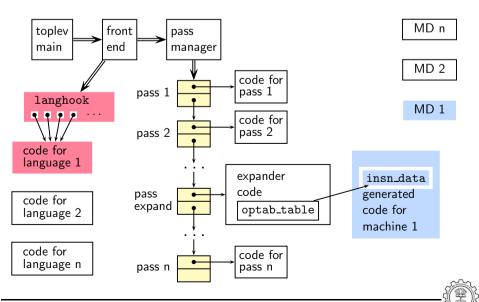
Retargetability Mechanism of GCC



Essential Abstractions in GCC GCC Resource Center, IIT Bombay

3 July 2011 Essential Abstrations: Summary 13/15

Plugin Structure in cc1



Retargetability Mechanism of GCC

Note

Essential Abstractions in GCC

GCC Resource Center, IIT Bombay

13/15

Essential Abstrations: Summary

Plugin Structure in cc1

Notes

3 July 2011



gcc

Essential Abstractions in GCC

14/15

cc1

as

lto1

as

ld

14/15

Essential Abstrations: Summary The GNU Tool Chain for LTO Support

Essential Abstractions in GCC

GCC Resource Center, IIT Bomb

14/15

Essential Abstrations: Summary The GNU Tool Chain for LTO Support

3 July 2011

GCC Resource Center, IIT Bombay

Single .o file + glibc/newlib

lto1'

lto1'

common "Fat" .s files as

"Fat" .o files

common Single .s file

ld

a.out file

cc1'

3 July 2011 **Essential Abstrations: Summary**

collect2

collect2

The GNU Tool Chain for LTO Support

Essential Abstrations: Summary

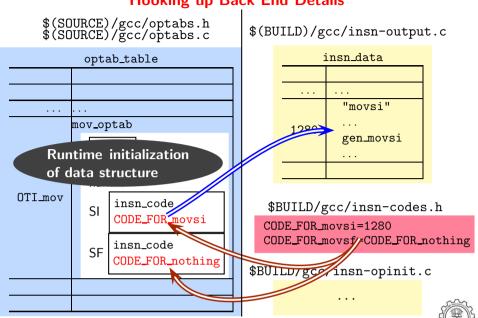
The GNU Tool Chain for LTO Support

```
lto1
                                                           cc1
 Common Code (executed twice for each function in the input program)
cgraph_optimize
  ipa_passes
      execute_ipa_pass_list(all_small_ipa_passes)/*!in lto*/
         execute_ipa_summary_passes(all_regular_ipa_passes)
         execute_ipa_summary_passes(all_lto_gen_passes)
         ipa_write_summaries
      cgraph_expand_all_functions
         cgraph_expand_function
         /* Intraprocedural passes on GIMPLE, */
         /* expansion pass, and passes on RTL. */
                                      a.out file
```





Essential Abstrations: Summary Hooking up Back End Details



Essential Abstractions in GCC

GCC Resource Center, IIT Bombay

Hooking up Back End Details

Note

